

Ein **Prepared Statement** ist eine sogenannte vorbereitete Anweisung für ein Datenbanksystem. Im Gegensatz zu gewöhnlichen Statements enthält es noch keine Parameterwerte. Stattdessen werden dem Datenbanksystem Platzhalter übergeben.

Mittels Prepared Statements können **SQL-Injections** effektiv verhindert werden, da das Datenbanksystem die Gültigkeit von Parametern prüft, bevor diese verarbeitet werden.



Quelle : https://de.wikipedia.org/wiki/Prepared_Statement (letzter Aufruf 4/2021)

Phase A: Vorbereitung des Prepared Statements		
Der erste Schritt besteht darin, dass ein Anweisungstemplate erzeugt wird. Anstelle der konkreten Werte für die relevanten Parameter werden die bereits erwähnten Platzhalter eingebaut, die man auch als Bind-Variablen bezeichnet. Typischerweise sind diese wie im folgenden Beispiel durch ein „?“ gekennzeichnet:		
SET @Statement= "INSERT INTO Tab(Attr1,Attr2) VALUES (? , ?)";		
Phase B: Verarbeitung des Anweisungsmusters durch das DBMS		
Vom Datenbankmanagementsystem wird das Anweisungsmuster zunächst geparkt (analysiert), damit es im nächsten Schritt kompiliert, also in eine ausführbare Anweisung umgewandelt werden kann. Bei diesem Vorgang wird das Prepared Statement außerdem optimiert.		
PREPARE stmt_name FROM @Statement;		
Phase C: Ausführung des Prepared Statements		
Zu einem späteren Zeitpunkt kann das verarbeitete Muster im Datenbanksystem beliebig häufig ausgeführt werden. Voraussetzung hierfür ist der passende Input, die die entsprechenden Werte für die Platzhalter-Felder liefern muss.		
SET @val1 = "Zeichenkette"; SET @val2 = numerischer_Wert; EXECUTE stmt_name USING @val1,@val2;		
Phase D: Freigeben (löschen) des Prepared Statements		
Am Ende wird eine mit PREPARE erzeugte vorbereitete Anweisung wieder freizugeben. Eine vorbereitete Anweisung wird implizit freigegeben, wenn ein neuer PREPARE-Befehl ausgegeben wird.		
DEALLOCATE PREPARE stmt_name ;		

Hinweis: Auch SELECT und UPDATE – Statements können als **Prepared Statement** verarbeitet werden.